Search Results - Record(s) 1 through 14 of 14 returned.

1. Document ID: US 20030144810 A1

L6: Entry 1 of 14 File: PGPB Jul 31, 2003

DOCUMENT-IDENTIFIER: US 20030144810 A1
TITLE: Methods and apparatus for data analysis

Detail Description Paragraph (37):
[0069] The smoothing system suitably operates by initially adjusting an initial value of a selected tester datum according to a first smoothing technique, and supplementarily adjusting the value according to a second smoothing technique if at least one of the initial value and the initially adjusted value meets a threshold. The first smoothing technique tends to smooth the data. The second smoothing technique also tends to smooth the data and/or improve tracking of the data, but in a different manner from the first smoothing technique. Further, the threshold may comprise any suitable criteria for determining whether to apply supplemental smoothing. The smoothing system suitably compares a plurality of preceding adjusted data to a plurality of preceding raw data to generate a comparison result, and applies a second smoothing technique to the selected datum to adjust the value of the selected datum according to whether the comparison result meets a first threshold. Further, the smoothing system suitably calculates a predicted value of the selected datum, and may apply a third smoothing technique to the selected datum to adjust the value of the selected datum according to whether the predicted value meets a second threshold.

Detail Description Paragraph (54):
[0084] The smoothing system predicts a value of the current smoothed data point according to the calculated slope. The system then compares the difference between the previously calculated value for the current smoothed data point (S.sub.n) to the predicted value for the current smoothed data point to a range number (R.sub.3) (step 818). If the difference is greater than the range R.sub.3, then the occurrence may be noted and the current smoothed data point is not adjusted. If the difference is within the range R.sub.3, then the current smoothed data point is set equal to the difference between the calculated current smoothed data point (S.sub.n) and the predicted value for the current smoothed data point (S.sub.n-pred) multiplied by a third multiplier M.sub.3 and added to the original value of the current smoothed data point (step 820). The equation:

Detail Description Paragraph (56):
[0085] Thus, the current smoothed data point is set according to a modified difference between the original smoothed data point and the predicted smoothed data point, but reduced by a certain amount (when M.sub.3 is less than 1). Applying the predictive smoothing tends to reduce point-to-point noise sensitivity during relatively flat (or otherwise non-trending) portions of the signal. The limited application of the predictive smoothing process to the smoothed data points ensures that the calculated average based on the slope does not affect the smoothed data when significant changes are occurring in the raw data, i.e., when the raw data signal is not relatively flat.

Detail Description Paragraph (101):
[0118] Supplemental data analysis at the cell controller level tends to increase quality control at the probe, and this final test yields. In addition, quality issues may be identified at product run time, not later. Furthermore, the supplemental data analysis and signature analysis tends to improve the quality of data provided to the downstream and offline analysis tools, as well as test engineers or other personnel, by identifying outliers. For example, the computer 108 may include information on the wafer map identifying a group of components having signature analyses indicating a fault in the manufacturing process. Thus, the signature analysis system may identify potentially defective goods that went undetected using conventional test analysis.

2. Document ID: US 20030023170 A1

L6: Entry 2 of 14 File: PGPB Jan 30, 2003

DOCUMENT-IDENTIFIER: US 20030023170 A1
TITLE: Optically similar reference samples and related methods for multivariate calibration models used in optical spectroscopy

Summary of Invention Paragraph (10):
[0009] In the prediction step, the infrared light is coupled to a sample of unknown characteristic value, and the calibration model is applied to the original or transformed intensity variations of the appropriate wavelengths of light measured from this unknown sample. The result of the prediction step is the estimated value of the characteristic of the unknown sample.

Summary of Invention Paragraph (24):
[0023] To the extent that the methods described above use background samples, they do not use optically similar background samples to help maintain multivariate calibrations. An optically similar reference sample is a sample that optically interacts with the optical measurement system in a manner that simulates to a desired degree the optical interaction between the optical system and the test sample. One component of optical similarity is the creation of a spectral absorbance at selected wavelengths that is similar to the test sample. The result is similarly shaped spectra at these wavelengths for both the reference and measurement samples. To obtain a similar shape and matched average absorbance, the optically similar reference sample should absorb the same or similar intensity of light at each selected wavelength over the range of wavelengths measured. An optically dissimilar reference sample is a sample that optically interacts with the optical measurement system in a manner that does not adequately represent the instrument or environmental state. When a dissimilar reference is used, it generally consists of either air or the solvent in which the analyte of interest is dissolved (e.g., an empty sample holder). In cases where the spectral signature of the analyte of interest is large compared to the spectral features due to any other component in the system, the use of an empty sample holder may be sufficient to maintain a stable calibration model over time. Calibration in this instance is typically implemented by forming the ratio of the transmission spectrum of the unknown sample to the transmission spectrum of the reference sample. However, in cases where the spectral signature of the analyte of interest is much smaller than that of the other system components (e.g., glucose levels in blood or other aqueous solution), an empty sample holder or any other sample that is optically different from the prediction sample is not sufficient as a reference sample and is not effective for maintaining calibration.

Detail Description Paragraph (25):
[0082] Simulated results are presented for the effects of water vapor level variation on the in vitro measurement of glucose in reflectance using scattering media. Actual spectra from 98 glucose solution samples were collected using an FTIR spectrometer operated at 16 cm.sup.-1 resolution. The samples contained variable levels of scattering media to simulate optical pathlength distributions similar to those seen in living tissue. For comparison purposes, spectra from two different types of background samples were also collected: a similar background with matched optical properties and an air background (i.e. an integrating sphere placed over the reflectance sampler). High-resolution water vapor spectra (obtained at 1 cm.sup.-1) were then artificially added to the solution and background spectra in order to simulate varying water vapor levels. Simulations were run on the resulting spectra in order to model the effects of finite instrument resolution on the added interferents. The sample spectra were then ratioed to the background sample spectra in an attempt to remove the effects of the varying water vapor levels. FIG. 1 shows the residual spectral effects after this background correction was performed. The two plots in FIG. 1 show the remaining spectral differences when the ratioed spectra with added water vapor are subtracted from the original ratioed spectra without added water vapor. As can be seen in the figure, the spectrally similar background reduces the effects of the water vapor interferent by a significant amount. A calibration developed at a constant water vapor was used to predict on the sample spectra. As stated above, the sample spectra were ratioed against a similar background with matched optical properties and an air background. The prediction errors for the sample data with the air background ratio were inflated over the sample spectra with a similar background by approximately 40 mg/dl using a calibration model with 20 factors. This simulation clearly demonstrates the advantage of using a similar background for correcting for even simple system perturbations.

☐ 3. Document ID: US 20030014205 A1

L6: Entry 3 of 14 File: PGPB Jan 16, 2003

DOCUMENT-IDENTIFIER: US 20030014205 A1
TITLE: Methods and apparatus for semiconductor testing

Detail Description Paragraph (38):
[0059] The smoothing system suitably operates by initially adjusting an initial value of a selected tester datum according to a first smoothing technique, and supplementarily adjusting the value according to a second smoothing technique if at least one of the initial value and the initially adjusted value meets a threshold. The first smoothing technique tends to smooth the data. The second smoothing technique also tends to smooth the data and/or improve tracking of the data, but in a different manner from the first smoothing technique. Further, the threshold may comprise any suitable criteria for determining whether to apply supplemental smoothing. The smoothing system suitably compares a plurality of preceding adjusted data to a plurality of preceding raw data to generate a comparison result, and applies a second smoothing technique to the selected datum to adjust the value of the selected datum according to whether the comparison result meets a first threshold. Further, the smoothing system suitably calculates a predicted value of the selected datum, and may apply a third smoothing technique to the selected datum to adjust the value of the selected datum according to whether the predicted value meets a second threshold.

Detail Description Paragraph (55):
[0074] The smoothing system predicts a value of the current smoothed data point according to the calculated slope. The system then compares the difference between the previously calculated value for the current smoothed data point (S.sub.n) to the predicted value for the current smoothed data point to a range number (R.sub.3) (step 818). If the difference is greater than the range R.sub.3, then the occurrence may be noted and the current smoothed data point is not adjusted. If the difference is within the range R.sub.3, then the current smoothed data point is set equal to the difference between the calculated current smoothed data point (S.sub.n) and the predicted value for the current smoothed data point (S.sub.n-pred) multiplied by a third multiplier M.sub.3 and added to the original value of the current smoothed data point (step 820). The equation:

Detail Description Paragraph (57):
[0075] Thus, the current smoothed data point is set according to a modified difference between the original smoothed data point and the predicted smoothed data point, but reduced by a certain amount (when M.sub.3 is less than 1). Applying the predictive smoothing tends to reduce point-to-point noise sensitivity during relatively flat (or otherwise nontrending) portions of the signal. The limited application of the predictive smoothing process to the smoothed data points ensures that the calculated average based on the slope does not affect the smoothed data when significant changes are occurring in the raw data, i.e., when the raw data signal is not relatively flat.

Detail Description Paragraph (99):
[0105] Supplemental data analysis at the cell controller level tends to increase quality control at the probe, and this final test yields. In addition, quality issues may be identified at product run time, not later. Furthermore, the supplemental data analysis and signature analysis tends to improve the quality of data provided to the downstream and offline analysis tools, as well as test engineers or other personnel, by identifying outliers. For example, the computer 108 may include information on the wafer map identifying a group of components having signature analyses indicating a fault in the manufacturing process. Thus, the signature analysis system may identify potentially defective goods that went undetected using conventional test analysis.

☐ 4. Document ID: US 6631470 B2

L6: Entry 4 of 14 File: USPT Oct 7, 2003

DOCUMENT-IDENTIFIER: US 6631470 B2
TITLE: Block based design methodology

Detailed Description Text (88):
There are three different classes of experience-based data used in design, each form of data being associated with a specific error profile: a) Project Data--Designer-requested estimate at project time. The designer does not draw upon the experience of others as logged in the FOE database, but more upon his own uncatalogued design experience. Error in the design estimate is given by a Designer-Error Variance, which has been observed for general designs. Designer-Error Variance is built from measuring a general history of designers' ability to accurately predict results. b) Predicted Data--Within a design classification but without a specific project in mind, a designer is requested to give his best-guess parameter-relationships for extending existing FOE data. In this case, the FOE data being extended may consist of as little as a single design-point. Error for this is in part specified by the designer's best guess at the parameterization error, but also modified by the history of designers' ability to accurately predict results. Assuming statistical independence, these error variances would be summed. c) Collated Data--Collected, classified and parameterized data from a set of design experiences. There is a possibility of measurement error directly associated with this data, but this is likely to be minor. The main error is defined as the difference between measured results and those predicted by the variation of data-parameters.

Detailed Description Text (90):
Predicted Data are referred to as FOE seed-data. Predicted Data may be immediately applied to FOE estimation on like designs.

Detailed Description Text (92):
The parameters listed above are used to extrapolate from existing, general FOE data to derive project-specific FOE estimates. Such a relationship between extrapolated estimates and FOE data is preferably defined for each design classification. Each parameter FOE relationship may be defined by a designer's personal experience (see Predicted Data above), or may be empirically specified through curve-fitting the FOE data if sufficient information is available. Parameters might include such technical variables as pipeline depth, degree of parallelism, bit-width, and clocking-speed.

Detailed Description Text (286):
Because the chip-level DFT architecture has only a single level, all attributes are at the top level. It is therefore intended that the designer should use the following architectural rules in accordance with the method of the present invention to put attributes in extractable comment form in the top-level design file: 1. Describe the DFT architecture hierarchically. 2. Create a single chip access port (CAP) at the highest level of hierarchy. The CAP specification should preferably: a. Map all test control and test data pins to the package-level pin to consistently maintain design and test data. b. Separate the test control pins from the test data pins. c. Set the test control pin attribute to either dedicated or selectable: i. dedicated if it should preferably be exclusively deactivated in normal mode; a dedicated pin cannot be shared with a functional pin. ii. selectable if it can be set to a test constant--a logical value--throughout a test; a selectable pin can be shared with a functional pin. d. Set the test data pin attribute to: test_clock if it is used as a clock during test; a test_clock pin can only be shared with an external functional clock pin. test_async if it is used asynchronously during test for reset; a test_async pin can be dedicated or shared if it does not cause any conflicts with other tests, test modes, or isolation modes. test_group(i) where (i) is the test_clock with which the test_group pin is synchronized during a test. e. Describe the following for each test mode: i. The test setup needed to gain access to the device under test if it requires an accessing sequence. Describe the protocol, such as JTAG instruction, test clock, or test reset. ii. The test execution needed to perform the actual test. Describe the test sequence in phases down to the task level, the iteration counts, the cycle time, the test length, and the test results. iii. The test postprocessing needed to close out the test and put the chip back in the default condition (normal mode). 3. Create a CAP controller specification that describes the test setup and test processing sequences for each test mode. The specification should preferably be implementable (synthesizable) and verifiable (via test benches and test sequences). 4. The designer may optionally specify a set of staging latches to fold the internal test data bus into the available test data pins. The staging action should preferably not alter the subsequent test result. The staging should preferably be a. Free from state-altering, time-sensitive signals. Use test_async signals or follow the persistent order of occurrence relative to the test_clock to resolve it. b. If it is not free from state-altering, time-sensitive signals, it should have extra test pins. This rule should preferably be used judiciously to avoid test

packaging problems. 5. The designer may optionally specify a test data signature analysis capability such as MISR to compress the test data, which minimizes the physical I/O constraint. The signature analysis should preferably be deterministic for each cycle of operation and should preferably: a. be free from X-value propagation by avoiding it at the MISR inputs. b. if step a. fails, suppress the affected MISR cycle. This rule should be followed judiciously to avoid the loss of fault coverage. 6. The designer may optionally create a set of other test mechanisms at the chip periphery to perform the following special tests: DC and AC parametric tests such as boundary scan tests; frequency tests such as PLL tests; and mixed-signal tests such as ADO and DAC tests. The control pins for these tests should preferably be included in the table of all test_control pins. The designer might also want to include them in the CAP controller specification to avoid conflicting interactions. 7. Specify a single device access port (DAP) at the next level of hierarchy, the level without I/Os or I/O-related cells, unrestricted to the physical I/O. 8. The DAP should preferably be a hybridized test port that can be formed by concatenating, merging, resizing, and multiplexing generic ports, such as TAP-based ports. 9. The designer should preferably be able to configure the DAP directly from the CAP controller. Partition each configuration into test control, test data, or test isolation ports. In each configuration: a. Set the test control port attribute to test_con f(k) if it should preferably be used to set the targeted configuration k. test_select if it can be set to a test constant. b. Set the test data port attribute to test_clock if it is used as a clock during test. test_async if it is used asynchronously during test. test_group(i) where (I) indicates the test clock to which the ports are synchronized. test_direction if it is used to indicate the test data direction. The test direction can only be a 1or 0 value. c. Set the test isolation port attribute to safe_state if it should preferably be isolated during test with a safe state logic value of 0,1, or Z, and to dont_care if it can be set to a non-floating logic value of 0 or 1. 10. Specify the interconnection of the CAP, the CAP controller, the staging latches, the MISR, the DAP, and the other test mechanisms 11. Specify the CAP controller, the staging latches, the MISR, the design body, and the other test mechanisms in a dedicated section. 12. Specify detail on the DAP the sockets, the UDL, and the test interconnect for the design body architecture only. 13. The design body architecture should preferably be described hierarchically. 14. There should preferably be multiple SAPs at the next level of hierarchy, the socket level. 15. Each SAP should preferably be a recursive image of the DAP with one or many applicable configurations available to the DAP. All configurations of the SAP should preferably be supported by the DAR.

☐ 5. Document ID: US 6629293 B2

L6: Entry 5 of 14 File: USPT Sep 30, 2003

DOCUMENT-IDENTIFIER: US 6629293 B2
TITLE: Block based design methodology

Detailed Description Text (86):
There are three different classes of experience-based data used in design, each form of data being associated with a specific error profile: a) Project Data--Designer-requested estimate at project time. The designer does not draw upon the experience of others as logged in the FOE database, but more upon his own uncatalogued design experience. Error in the design estimate is given by a Designer-Error Variance, which has been observed for general designs. Designer-Error Variance is built from measuring a general history of designers' ability to accurately predict results. b) Predicted Data--Within a design classification but without a specific project in mind, a designer is requested to give his best-guess parameter-relationships for extending existing FOE data. In this case, the FOE data being extended may consist of as little as a single design-point. Error for this is in part specified by the designer's best guess at the parameterization error, but also modified by the history of designers' ability to accurately predict results. Assuming statistical independence, these error variances would be summed. c) Collated Data--Collected, classified and parameterized data from a set of design experiences. There is a possibility of measurement error directly associated with this data, but this is likely to be minor. The main error is defined as the difference between measured results and those predicted by the variation of data-parameters.

Detailed Description Text (88):
Predicted Data are referred to as FOE seed-data. Predicted Data may be immediately applied to FOE estimation on like designs.

Detailed Description Text (90):

The parameters listed above are used to extrapolate from existing, general FOE data to derive project-specific FOE estimates. Such a relationship between extrapolated estimates and FOE data is preferably defined for each design classification. Each parameter FOE relationship may be defined by a designer's personal experience (see Predicted Data above), or may be empirically specified through curve-fitting the FOE data if sufficient information is available. Parameters might include such technical variables as pipeline depth, degree of parallelism, bit-width, and clocking-speed.

Detailed Description Text (279):
Because the chip-level DFT architecture has only a single level, all attributes are at the top level. It is therefore intended that the designer should use the following architectural rules in accordance with the method of the present invention to put attributes in extractable comment form in the top-level design file: 1. Describe the DFT architecture hierarchically. 2. Create a single chip access port (CAP) at the highest level of hierarchy. The CAP specification should preferably: a. Map all test control and test data pins to the package-level pin to consistently maintain design and test data. b. Separate the test control pins from the test data pins. c. Set the test control pin attribute to either dedicated or selectable: i. dedicated if it should preferably be exclusively deactivated in normal mode; a dedicated pin cannot be shared with a functional pin. ii. selectable if it can be set to a test constant--a logical value--throughout a test; a selectable pin can be shared with a functional pin. d. Set the test data pin attribute to: test_clock if it is used as a clock during test; a test_clock pin can only be shared with an external functional clock pin. test_async if it is used asynchronously during test for reset; a test_async pin can be dedicated or shared if it does not cause any conflicts with other tests, test modes, or isolation modes. test_group(i) where (i) is the test_clock with which the test_group pin is synchronized during a test. e. Describe the following for each test mode: i. The test setup needed to gain access to the device under test if it requires an accessing sequence. Describe the protocol, such as JTAG instruction, test clock, or test reset. ii. The test execution needed to perform the actual test. Describe the test sequence in phases down to the task level, the iteration counts, the cycle time, the test length, and the test results. iii. The test postprocessing needed to close out the test and put the chip back in the default condition (normal mode). 3. Create a CAP controller specification that describes the test setup and test processing sequences for each test mode. The specification should preferably be implementable (synthesizable) and verifiable (via test benches and test sequences). 4. The designer may optionally specify a set of staging latches to fold the internal test data bus into the available test data pins. The staging action should preferably not alter the subsequent test result. The staging should preferably be a. Free from state-altering, time-sensitive signals. Use test_async signals or follow the persistent order of occurrence relative to the test_clock to resolve it. b. If it is not free from state-altering, time-sensitive signals, it should have extra test pins. This rule should preferably be used judiciously to avoid test packaging problems. 5. The designer may optionally specify a test data signature analysis capability such as MISR to compress the test data, which minimizes the physical I/O constraint. The signature analysis should preferably be deterministic for each cycle of operation and should preferably: a. be free from X-value propagation by avoiding it at the MISR inputs. b. if step a. fails, suppress the affected MISR cycle. This rule should be followed judiciously to avoid the loss of fault coverage. 6. The designer may optionally create a set of other test mechanisms at the chip periphery to perform the following special tests: DC and AC parametric tests such as boundary scan tests; frequency tests such as PLL tests; and mixed-signal tests such as ADO and DAC tests. The control pins for these tests should preferably be included in the table of all test_control pins. The designer might also want to include them in the CAP controller specification to avoid conflicting interactions. 7. Specify a single device access port (DAP) at the next level of hierarchy, the level without I/Os or I/O-related cells, unrestricted to the physical I/O. 8. The DAP should preferably be a hybridized test port that can be formed by concatenating, merging, resizing, and multiplexing generic ports, such as TAP-based ports. 9. The designer should preferably be able to configure the DAP directly from the CAP controller. Partition each configuration into test control, test data, or test isolation ports. In each configuration: a. Set the test control port attribute to test_con f(k) if it should preferably be used to set the targeted configuration k. test_select if it can be set to a test constant. b. Set the test data port attribute to test_clock if it Is used as a clock during test. test_async if it is used asynchronously during test. test_group(i) where (I) indicates the test clock to which the ports are synchronized. test_direction if it is used to indicate the test data direction. The test direction can only be a 1 or 0 value. c. Set the test isolation port attribute to safe_state if it should preferably be isolated during test with a safe state logic value of 0, 1, or Z, and to dont_care if it can be set to a non-floating logic value of 0 or 1. 10. Specify the interconnection of the CAP, the CAP controller, the staging latches, the MISR, the DAP, and the other test mechanisms 11. Specify the CAP controller, the staging latches, the MISR, the design body, and the other test mechanisms in a dedicated section. 12. Specify detail on the DAP the sockets, the UDL, and the test interconnect for the design body architecture only. 13. The design body architecture should preferably be described hierarchically. 14. There should preferably be multiple SAPs at the next level of hierarchy, the socket level. 15. Each SAP should preferably be a recursive image of the DAP with one or many applicable configurations available to the DAP. All configurations of the SAP should preferably be supported by the DAR.

6. Document ID: US 6594800 B2

L6: Entry 6 of 14 File: USPT Jul 15, 2003

DOCUMENT-IDENTIFIER: US 6594800 B2
TITLE: Block based design methodology

Detailed Description Text (106):
Designer-requested estimate at project time. The designer does not draw upon the experience of others as logged in the FOE database, but more upon his own uncatalogued design experience. Error in the design estimate is given by a Designer-Error Variance, which has been observed for general designs. Designer-Error Variance is built from measuring a general history of designers' ability to accurately predict results. b) Predicted Data

Detailed Description Text (108):
Collected, classified and parameterized data from a set of design experiences. There is a possibility of measurement error directly associated with this data, but this is likely to be minor. The main error is defined as the difference between measured results and those predicted by the variation of data-parameters.

Detailed Description Text (110):
Predicted Data are referred to as FOE seed-data. Predicted Data may be immediately applied to FOE estimation on like designs.

Detailed Description Text (112):
The parameters listed above are used to extrapolate from existing, general FOE data to derive project-specific FOE estimates. Such a relationship between extrapolated estimates and FOE data is preferably defined for each design classification. Each parameter FOE relationship may be defined by a designer's personal experience (see Predicted Data above), or may be empirically specified through curve-fitting the FOE data if sufficient information is available. Parameters might include such technical variables as pipeline depth, degree of parallelism, bit-width, and clocking-speed.

Detailed Description Text (308):
Because the chip-level DFT architecture has only a single level, all attributes are at the top level. It is therefore intended that the designer should use the following architectural rules in accordance with the method of the present invention to put attributes in extractable comment form in the top-level design file: 1. Describe the DFT architecture hierarchically. 2. Create a single chip access port (CAP) at the highest level of hierarchy. The CAP specification should preferably: a. Map all test control and test data pins to the package-level pin to consistently maintain design and test data. b. Separate the test control pins from the test data pins. c. Set the test control pin attribute to either dedicated or selectable: i. dedicated if it should preferably be exclusively deactivated in normal mode; a dedicated pin cannot be shared with a functional pin. ii. selectable if it can be set to a test constant--a logical value--throughout a test; a selectable pin can be shared with a functional pin. d. Set the test data pin attribute to: test_clock if it is used as a clock during test; a test_clock pin can only be shared with an external functional clock pin. test_async if it is used asynchronously during test for reset; a test_async pin can be dedicated or shared if it does not cause any conflicts with other tests, test modes, or isolation modes. test_group(i) where (i) is the test_clock with which the test_group pin is synchronized during a test. e. Describe the following for each test mode: i. The test setup needed to gain access to the device under test if it requires an accessing sequence. Describe the protocol, such as JTAG instruction, test clock, or test reset. ii. The test execution needed to perform the actual test. Describe the test sequence in phases down to the task level, the iteration counts, the cycle time, the test length, and the test results. iii. The test postprocessing needed to close out the test and put the chip back in the default condition (normal mode). 3. Create a CAP controller specification that describes the test setup and test processing sequences for each test mode. The specification should preferably be implementable (synthesizable) and verifiable (via test benches and test sequences). 4. The designer may optionally specify a set of staging latches to fold the internal test data bus into the available test data pins. The staging action should preferably not alter the subsequent test result. The staging should preferably be a. Free from state-altering, time-sensitive signals. Use test_async signals or follow the persistent order of occurrence relative to the test_clock to resolve it. b. If it is not free from state-altering, time-sensitive signals, it should have extra test pins. This rule should preferably be used judiciously to avoid test

packaging problems. 5. The designer may optionally specify a test data signature analysis capability such as MISR to compress the test data, which minimizes the physical I/O constraint. The signature analysis should preferably be deterministic for each cycle of operation and should preferably: a. be free from X-value propagation by avoiding it at the MISR inputs. b. if step a. fails, suppress the affected MISR cycle. This rule should be followed judiciously to avoid the loss of fault coverage. 6. The designer may optionally create a set of other test mechanisms at the chip periphery to perform the following special tests: DC and AC parametric tests such as boundary scan tests; frequency tests such as PLL tests; and mixed-signal tests such as ADO and DAC tests. The control pins for these tests should preferably be included in the table of all test_control pins. The designer might also want to include them in the CAP controller specification to avoid conflicting interactions. 7. Specify a single device access port (DAP) at the next level of hierarchy, the level without I/Os or I/O-related cells, unrestricted to the physical I/O. 8. The DAP should preferably be a hybridized test port that can be formed by concatenating, merging, resizing, and multiplexing generic ports, such as TAP-based ports. 9. The designer should preferably be able to configure the DAP directly from the CAP controller. Partition each configuration into test control, test data, or test isolation ports. In each configuration: a. Set the test control port attribute to test_con f(k) if it should preferably be used to set the targeted configuration k. test_select if it can be set to a test constant. b. Set the test data port attribute to test_clock if it Is used as a clock during test. test_async if it is used asynchronously during test. test_group(i) where (I) indicates the test clock to which the ports are synchronized. test_direction if it is used to indicate the test data direction. The test direction can only be a 1 or 0 value. c. Set the test isolation port attribute to safe_state if it should preferably be isolated during test with a safe state logic value of 0,1, or Z, and to dont_care if it can be set to a non-floating logic value of 0 or 1. 10. Specify the interconnection of the CAP, the CAP controller, the staging latches, the MISR, the DAP, and the other test mechanisms 11. Specify the CAP controller, the staging latches, the MISR, the design body, and the other test mechanisms in a dedicated section. 12. Specify detail on the DAP the sockets, the UDL, and the test interconnect for the design body architecture only. 13. The design body architecture should preferably be described hierarchically. 14. There should preferably be multiple SAPs at the next level of hierarchy, the socket level. 15. Each SAP should preferably be a recursive image of the DAP with one or many applicable configurations available to the DAP. All configurations of the SAP should preferably be supported by the DAR.

7. Document ID: US 6574778 B2

L6: Entry 7 of 14 File: USPT Jun 3, 2003

DOCUMENT-IDENTIFIER: US 6574778 B2
TITLE: Block based design methodology

Detailed Description Text (88):
There are three different classes of experience-based data used in design, each form of data being associated with a specific error profile: a) Project Data--Designer-requested estimate at project time. The designer does not draw upon the experience of others as logged in the FOE database, but more upon his own uncatalogued design experience. Error in the design estimate is given by a Designer-Error Variance, which has been observed for general designs. Designer-Error Variance is built from measuring a general history of designers' ability to accurately predict results. b) Predicted Data--Within a design classification but without a specific project in mind, a designer is requested to give his best-guess parameter-relationships for extending existing FOE data. In this case, the FOE data being extended may consist of as little as a single design-point. Error for this is in part specified by the designer's best guess at the parameterization error, but also modified by the history of designers' ability to accurately predict results. Assuming statistical independence, these error variances would be summed. c) Collated Data--Collected, classified and parameterized data from a set of design experiences. There is a possibility of measurement error directly associated with this data, but this is likely to be minor. The main error is defined as the difference between measured results and those predicted by the variation of data-parameters.

Detailed Description Text (90):
Predicted Data are referred to as FOE seed-data. Predicted Data may be immediately applied to FOE estimation on like designs.

Detailed Description Text (92):

The parameters listed above are used to extrapolate from existing, general FOE data to derive project-specific FOE estimates. Such a relationship between extrapolated estimates and FOE data is preferably defined for each design classification. Each parameter FOE relationship may be defined by a designer's personal experience (see Predicted Data above), or may be empirically specified through curve-fitting the FOE data if sufficient information is available. Parameters might include such technical variables as pipeline depth, degree of parallelism, bit-width, and clocking-speed.

Detailed Description Text (304):
Because the chip-level DFT architecture has only a single level, all attributes are at the top level. It is therefore intended that the designer should use the following architectural rules in accordance with the method of the present invention to put attributes in extractable comment form in the top-level design file: 1. Describe the DFT architecture hierarchically. 2. Create a single chip access port (CAP) at the highest level of hierarchy. The CAP specification should preferably: a. Map all test control and test data pins to the package-level pin to consistently maintain design and test data. b. Separate the test control pins from the test data pins. c. Set the test control pin attribute to either dedicated or selectable: i. dedicated if it should preferably be exclusively deactivated in normal mode; a dedicated pin cannot be shared with a functional pin. ii. selectable if it can be set to a test constant a logical value--throughout a test; a selectable pin can be shared with a functional pin. d. Set the test data pin attribute to: test_clock if it is used as a clock during test; a test_clock pin can only be shared with an external functional clock pin. test_async if it is used asynchronously during test for reset; a test_async pin can be dedicated or shared if it does not cause any conflicts with other tests, test modes, or isolation modes. test_group (i) where (ii) is the test_clock with which the test_group pin is synchronized during a test. e. Describe the following for each test mode: i. The test setup needed to gain access to the device under test if it requires an accessing sequence. Describe the protocol, such as JTAG instruction, test clock, or test reset. ii. The test execution needed to perform the actual test. Describe the test sequence in phases down to the task level, the iteration counts, the cycle time, the test length, and the test results. iii. The test postprocessing needed to close out the test and put the chip back in the default condition (normal mode). 3. Create a CAP controller specification that describes the test setup and test processing sequences for each test mode. The specification should preferably be implementable (synthesizable) and verifiable (via test benches and test sequences). 4. The designer may optionally specify a set of staging latches to fold the internal test data bus into the available test data pins. The staging action should preferably not alter the subsequent test result. The staging should preferably be a. Free from state-altering, time-sensitive signals. Use test_async signals or follow the persistent order of occurrence relative to the test_clock to resolve it. b. If it is not free from state-altering, time-sensitive signals, it should have extra test pins. This rule should preferably be used judiciously to avoid test packaging problems. 5. The designer may optionally specify a test data signature analysis capability such as MISR to compress the test data, which minimizes the physical I/O constraint. The signature analysis should preferably be deterministic for each cycle of operation and should preferably: a. be free from X-value propagation by avoiding it at the MISR inputs. b. if step a. fails, suppress the affected MISR cycle. This rule should be followed judiciously to avoid the loss of fault coverage. 6. The designer may optionally create a set of other test mechanisms at the chip periphery to perform the following special tests: DC and AC parametric tests such as boundary scan tests; frequency tests such as PLL tests; and mixed-signal tests such as ADO and DAC tests. The control pins for these tests should preferably be included in the table of all test_control pins. The designer might also want to include them in the CAP controller specification to avoid conflicting interactions. 7. Specify a single device access port (DAP) at the next level of hierarchy, the level without I/Os or I/O-related cells, unrestricted to the physical I/O. 8. The DAP should preferably be a hybridized test port that can be formed by concatenating, merging, resizing, and multiplexing generic ports, such as TAP-based ports. 9. The designer should preferably be able to configure the DAP directly from the CAP controller. Partition each configuration into test control, test data, or test isolation ports. In each configuration: a. Set the test control port attribute to test_con f(k) if it should preferably be used to set the targeted configuration k. test_select if it can be set to a test constant. b. Set the test data port attribute to test_clock if it Is used as a clock during test. test_async if it is used asynchronously during test. test_group (i) where (I) indicates the test clock to which the ports are synchronized. test_direction if it is used to indicate the test data direction. The test direction can only be a 1 or 0 value. c. Set the test isolation port attribute to safe_state if it should preferably be isolated during test with a safe state logic value of 0, 1, or Z, and to dont_care if it can be set to a non-floating logic value of 0 or 1. 10. Specify the interconnection of the CAP, the CAP controller, the staging latches, the MISR, the DAP, and the other test mechanisms 11. Specify the CAP controller, the staging latches, the MISR, the design body, and the other test mechanisms in a dedicated section. 12. Specify detail on the DAP the sockets, the UDL, and the test interconnect for the design body architecture only. 13. The design body architecture should preferably be described hierarchically. 14. There should preferably be multiple SAPs at the next level of hierarchy, the socket level. 15. Each SAP should preferably be a recursive image of the DAP with one or many applicable configurations available to the DAP. All configurations of the SAP should preferably be supported by the DAR.

CLAIMS:

5. The method of claim 1, wherein the identified experience data includes at least one of predicted experience data or collated experience data.

6. The method of claim 1, further comprising the step of: (f) building a model upon the identified designer's experience data to predict similar parameters of a similar design.

8. Document ID: US 6567957 B1

L6: Entry 8 of 14 File: USPT May 20, 2003

DOCUMENT-IDENTIFIER: US 6567957 B1
TITLE: Block based design methodology

Detailed Description Text (88):
There are three different classes of experience-based data used in design, each form of data being associated with a specific error profile: a) Project Data--Designer-requested estimate at project time. The designer does not draw upon the experience of others as logged in the FOE database, but more upon his own uncatalogued design experience. Error in the design estimate is given by a Designer-Error Variance, which has been observed for general designs. Designer-Error Variance is built from measuring a general history of designers' ability to accurately predict results. b) Predicted Data--Within a design classification but without a specific project in mind, a designer is requested to give his best-guess parameter-relationships for extending existing FOE data. In this case, the FOE data being extended may consist of as little as a single design-point. Error for this is in part specified by the designer's best guess at the parameterization error, but also modified by the history of designers' ability to accurately predict results. Assuming statistical independence, these error variances would be summed. c) Collated Data--Collected, classified and parameterized data from a set of design experiences. There is a possibility of measurement error directly associated with this data, but this is likely to be minor. The main error is defined as the difference between measured results and those predicted by the variation of data-parameters.

Detailed Description Text (90):
Predicted Data are referred to as FOE seed-data. Predicted Data may be immediately applied to FOE estimation on like designs. A common classification of the types of data received must apply to both of the above sources of FOE data. Such common classification permits the quick identification and cataloging of received data. Initial classification-specification is regarded as the planning stage for FOE, and the entering/gathering of data is the building stage. As the amount of information in the FOE database grows, the refinement process is applied to reduce error tolerances to within those being observed statistically. In parallel with all three of these stages is the FOE certification process.

Detailed Description Text (91):
The parameters listed above are used to extrapolate from existing, general FOE data to derive project-specific FOE estimates. Such a relationship between extrapolated estimates and FOE data is preferably defined for each design classification. Each parameter FOE relationship may be defined by a designer's personal experience (see Predicted Data above), or may be empirically specified through curve-fitting the FOE data if sufficient information is available. Parameters might include such technical variables as pipeline depth, degree of parallelism, bit-width, and clocking-speed.

Detailed Description Text (301):
Because the chip-level DFT architecture has only a single level, all attributes are at the top level. It is therefore intended that the designer should use the following architectural rules in accordance with the method of the present invention to put attributes in extractable comment form in the top-level design file: 1. Describe the DFT architecture hierarchically. 2. Create a single chip access port (CAP) at the highest level of hierarchy. The CAP specification should preferably: a. Map all test control and test data pins to the package-level pin to consistently maintain design and test data. b. Separate the test control pins from the test data pins. c. Set the test control pin attribute to either dedicated or selectable: i. dedicated if it should preferably be exclusively deactivated in normal mode; a dedicated

pin cannot be shared with a functional pin. ii. selectable if it can be set to a test constant--a logical value--throughout a test; a selectable pin can be shared with a functional pin. d. Set the test data pin attribute to: test_clock if it is used as a clock during test; a test_clock pin can only be shared with an external functional clock pin. test_async if it is used asynchronously during test for reset; a test_async pin can be dedicated or shared if it does not cause any conflicts with other tests, test modes, or isolation modes. test_group(i) where (i) is the test_clock with which the test_group pin is synchronized during a test. e. Describe the following for each test mode: i. The test setup needed to gain access to the device under test if it requires an accessing sequence. Describe the protocol, such as JTAG instruction, test clock, or test reset. ii. The test execution needed to perform the actual test. Describe the test sequence in phases down to the task level, the iteration counts, the cycle time, the test length, and the test results. iii. The test postprocessing needed to close out the test and put the chip back in the default condition (normal mode). 3. Create a CAP controller specification that describes the test setup and test processing sequences for each test mode. The specification should preferably be implementable (synthesizable) and verifiable (via test benches and test sequences). 4. The designer may optionally specify a set of staging latches to fold the internal test data bus into the available test data pins. The staging action should preferably not alter the subsequent test result. The staging should preferably be a. Free from state-altering, time-sensitive signals. Use test_async signals or follow the persistent order of occurrence relative to the test_clock to resolve it. b. If it is not free from state-altering, time-sensitive signals, it should have extra test pins. This rule should preferably be used judiciously to avoid test packaging problems. 5. The designer may optionally specify a test data signature analysis capability such as MISR to compress the test data, which minimizes the physical I/O constraint. The signature analysis should preferably be deterministic for each cycle of operation and should preferably: a. be free from X-value propagation by avoiding it at the MISR inputs. b. if step a. fails, suppress the affected MISR cycle. This rule should be followed judiciously to avoid the loss of fault coverage. 6. The designer may optionally create a set of other test mechanisms at the chip periphery to perform the following special tests: DC and AC parametric tests such as boundary scan tests; frequency tests such as PLL tests; and mixed-signal tests such as ADO and DAC tests. The control pins for these tests should preferably be included in the table of all test_control pins. The designer might also want to include them in the CAP controller specification to avoid conflicting interactions. 7. Specify a single device access port (DAP) at the next level of hierarchy, the level without I/Os or I/O-related cells, unrestricted to the physical I/O. 8. The DAP should preferably be a hybridized test port that can be formed by concatenating, merging, resizing, and multiplexing generic ports, such as TAP-based ports. 9. The designer should preferably be able to configure the DAP directly from the CAP controller. Partition each configuration into test control, test data, or test isolation ports. In each configuration: a. Set the test control port attribute to test_con f(k) if it should preferably be used to set the targeted configuration k. test_select if it can be set to a test constant. b. Set the test data port attribute to test_clock if it Is used as a clock during test. test_async if it is used asynchronously during test. test_group(i) where (I) indicates the test clock to which the ports are synchronized. test_direction if it is used to indicate the test data direction. The test direction can only be a 1 or 0 value. c. Set the test isolation port attribute to safe_state if it should preferably be isolated during test with a safe state logic value of 0, 1, or Z, and to dont_care if it can be set to a non-floating logic value of 0 or 1. 10. Specify the interconnection of the CAP, the CAP controller, the staging latches, the MISR, the DAP, and the other test mechanisms 11. Specify the CAP controller, the staging latches, the MISR, the design body, and the other test mechanisms in a dedicated section. 12. Specify detail on the DAP the sockets, the UDL, and the test interconnect for the design body architecture only. 13. The design body architecture should preferably be described hierarchically. 14. There should preferably be multiple SAPs at the next level of hierarchy, the socket level. 15. Each SAP should preferably be a recursive image of the DAP with one or many applicable configurations available to the DAP. All configurations of the SAP should preferably be supported by the DAR.

☐ 9. Document ID: US 6327684 B1

L6: Entry 9 of 14 File: USPT Dec 4, 2001

DOCUMENT-IDENTIFIER: US 6327684 B1
TITLE: Method of testing at-speed circuits having asynchronous clocks and controller for use therewith

Detailed Description Text (5):
Test controllers are generally well known in the art and, accordingly, the test controllers will not be described in detail except for the components thereof which are modified to achieve the objectives of the present invention. In

FIG. 1, each test controller is divided into two sections 26 and 28. Section 26 is a clock controller which operates at its domain base clock rate. Clock controller 30 is associated with clock domain 14. Clock controller 32 is associated with clock domain 16. Section 28 comprises other test controller sub-circuits which are driven by the coreclk signal output by the clock controller. Generally, the sub-circuits include a pattern generator 34 for generating test vectors or stimuli to be shifted into the scanable memory elements, a signature analyser 36 for analysing the results of a test, and a control signal generator 38 for generating appropriate signals at the appropriate times during test operations. The clock controller is shown separately from the other test controller sub-circuits because it is the only sub-circuit which needs to be modified to implement the method described herein. The other sub-circuits, may be the same as those known in the prior art.

Detailed Description Text (13):
It should be appreciated that even though FIG. 1 shows only one cross-boundary signal path between clock domains 14 and 16, many more such paths could exist. Signal paths can originate from any domain and terminate in any other domain. The signal paths could also traverse combinational logic (not shown). Since the clocks controlling the source and destination flip-flops are asynchronous with respect to each other, the time interval between the active edges of coreclock2 and coreclock1 is generally unknown and variable over time, the output of destination flip-flop of clock domain 14 normally cannot be predicted and the signature calculated by the signature analyser will not be repeatable.

10. Document ID: US 5862150 A

L6: Entry 10 of 14 File: USPT Jan 19, 1999

DOCUMENT-IDENTIFIER: US 5862150 A
TITLE: Video frame signature capture

Brief Summary Text (11):
Signature analysis is a well known method of hardware testing. It involves calculating a signature of a set of data, usually after this data has passed through some hardware under test. By comparing the actual signature with a known golden signature, a pass/fail determination of the hardware under test can be made.

Detailed Description Text (11):
Diagnostic software can use the signature analysis hardware to check that all of the pixels in a frame containing a known image have been correctly processed by the rendering controller, frame buffer and RAMDAC logic. After drawing a known image into the frame buffer, software asserts the signature capture "request" bit (signature analysis control register bit 25) to cause the RAMDAC signature analysis hardware to generate the signature for the image as it is scanned out of the frame buffer and into the digital-to-analog converters. Each different known image has a (nearly) unique signature, or `fingerprint`. As noted above, these signatures can be collected on a "known good" system, or predicted by simulation. Once determined, the "good" signatures can be used to test for the proper operation of manufactured copies of the system to ensure that the manufactured copies do not have defective parts. The signatures can also be used after a system has been installed as a diagnostic aid.

Detailed Description Text (14):
By utilizing this invention, each pixel of the frame contributes exactly once to the signature, and the contributions are in the correct raster scan order. This produces a deterministic signature for the image displayed in that frame, which can be used to detect and diagnose failures in the rendering controller 14, the VRAMs 15a-15d, the RAMDAC 21 digital logic and the interconnections among these parts. In almost all cases, the signature collected by the signature analysis hardware will not match the known good/predicted signature if even one bit in one pixel out of a million pixel image is incorrectly processed.

11. Document ID: US 5629870 A

L6: Entry 11 of 14 File: USPT May 13, 1997

DOCUMENT-IDENTIFIER: US 5629870 A
TITLE: Method and apparatus for predicting electric induction machine failure during operation

Brief Summary Text (4):
The present invention is directed to methods and apparatus which monitor power line waveform that is affected by an electrical device, such as for example an induction machine, during actual in-service operation of the machine in its working environment. Based upon analysis of the line waveform signature, the present invention predicts potential machine failure caused by defects in machine mechanical components, such as broken rotor bars and defective rolling element bearings which support the machine shaft.

Brief Summary Text (14):
Examining more closely the known electric induction machine direct mechanical vibration measurement and current harmonic analysis testing systems, each has strengths and weaknesses. Direct vibration measurement has the obvious advantage of measuring the harmful vibrations directly through the use of one or more piezoelectric vibration sensors. In addition, signal processing hardware is reduced since the signature to be analyzed need not be extracted from a much higher magnitude carrier signal, as in the case with current harmonic analysis. The disadvantage of the direct measurement is the fact that the machine must be outfitted with the sensor, and the sensor with its wiring is prone to failure. The current harmonic analysis has the advantage of not requiring any sensors in addition to those which already exist in currently available motor controllers. However, current harmonic analysis is computationally difficult since the harmonics in question are many times smaller in magnitude than the excitation frequency (i.e., 50 or 60 Hertz electric utility power). In addition, an understanding must be gained of the relationship between the current harmonics and machine vibrations and resultant motor electrical performance deterioration.

Detailed Description Text (4):
The purpose of the failure prediction system of the present invention is to monitor the excitation provided to an induction motor and from the motor current signature to predict the fact that the motor is likely to fail several days prior to a disabling failure. Production equipment systems desirably should be of moderate cost, comparable to that of commercially available electronic motor controllers and easy to install in existing industrial sites. It is contemplated that one prediction system may be designed to service several motors.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | KWIC | Draw Desc | Image

12. Document ID: US 5488615 A

L6: Entry 12 of 14 File: USPT Jan 30, 1996

DOCUMENT-IDENTIFIER: US 5488615 A
TITLE: Universal digital signature bit device

Detailed Description Text (9):
The theory of the occupancy of an LFSR used for signature analysis predicts that the probability of a single BIT error going undetected is equal to: ##EQU1##

Detailed Description Text (11):
For example, if the test sequence length (m) is limited so that it is equal to or greater than 256, and the digital signature BIT device is configured to have a signature analyzer length (n) equal to or greater than 12, then: ##EQU2##

13. Document ID: US 4551837 A

L6: Entry 13 of 14 File: USPT Nov 5, 1985

DOCUMENT-IDENTIFIER: US 4551837 A
TITLE: High speed operational recurring signature evaluator for digital equipment tester

Brief Summary Text (9):
A recent approach has used the concept of signature analysis based on use of a data compression technique to provide a unique fingerprint of each interconnection or test node in the device being tested. Signature analysis converts a serial bit stream into a unique multi-bit stream for a particular test node, the bit stream being started and stopped in a carefully controlled time window. When the signatures are recorded at each node of a correctly operating system, they can be compared to those from a unit under test. A difference in signatures indicates that the node is not operating as expected. If the test windows have been correctly defined, the signatures are totally characteristic of their respective test nodes.

Detailed Description Text (54):
The primary function of the Z80 based controller (FIG. 3) is to handle the task of configuring the data compressor for each node which will be probed when the host logic tester (PSP) is in guided probe mode. Data to the logic tester printer is the source for letting the controller know what node the tester will probe next. When a node name is printed, the controller looks in the test program ROM 316 for instructions stored under that name. The instructions tell the controller how to set up the data compressor and how to handle the signature data after it is taken. The signature may be transferred to the host logic tester serially from port 308 (terminal 318) or as a parallel 19 bit word from 220. The signature may be altered or replaced, depending on instructions to evaluate the signature originally taken, to tailor it to be what the host logic tester expects to receive. Special operations such as repeatedly retaking a signature a certain number of times and then checking for the occurrence of one or more predicted signatures can be done. In this case the number of occurrences of one of the predicted signatures may be specified by the data under the node name in the test program ROM 316.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | KWIC | Draw Desc | Image

14. Document ID: JP 01010184 A

L6: Entry 14 of 14 File: JPAB Jan 13, 1989

DOCUMENT-IDENTIFIER: JP 01010184 A
TITLE: MULTICHIP PACKAGING STRUCTURE AND METHOD FOR TESTING IT

Abstract Text (2):
CONSTITUTION: A chip (UUT) to be tested is discriminated from a module. The interference between all other chips from which the UUT is separated and tests is electrically inhibited. The count of all patterns applied upon the UUT is fetched from a memory. A set of pseudo-random or weighted pseudo- random patterns is propagated through the logic circuit of the UUT and becomes a predicted binary value which is found from the output of the UUT. The binary value is outputted from the UUT and accumulated in a signature analyzer and the accumulated result is sent to a comparator. A signature simulated for the UUT is inputted to the comparator from the memory and compared with an actual signature. Therefore, all chips of the module are tested and, when the diagnosis is completed, the process is stopped.

Generate Collection | Print

Terms	Documents
L3 and I2	14

Display Format: - Change Format

Previous Page Next Page